

有効数字の取扱いを考慮した数値計算のための コンピューター・プログラム

植 村 允 勝*

Computer Programs for the Numerical Calculations with the Proper Treatment of Significant Figures.

Masakatsu UYEMURA

Average freshmen are rather unfamiliar with the calculations of the data bearing experimental error. Such skills are, however, indispensable in the engineering courses. Some computer programs have been prepared to facilitate the freshmen's exercise on this matter. The programs perform calculations, counting the number of significant figures in the data and rounding digit at the correct place. The principles and flow charts of the programs are presented, and the performance of the programs are examined.

1. はじめに

大学の教養課程における「化学実験」などの実験科目では、学生に対して誤差の概念とその取扱いについて正しい認識を与えることがひとつの重要な課題となっており、その具体的内容としては、

- 1) 実験値は必ず誤差を含むこと.
- 2) 実験値に含まれる誤差をどのように表わすか.
- 3) 誤差を含む数値を使って得た計算結果の中には、もとの誤差の影響がどのように伝達されているか.

等の事項を理解させて、これらを応用させることなどが含まれる.

このような事柄が大学の教養課程の教育において重要な課題となる背景として次のようなことが挙げられる. まず第1に、高校課程の化学では、化学の基礎的概念や原理について、かなり詳しく

扱われており、かつ実験についても触れられているのだが、実験の定量的な問題にあまり力点が置かれていないということである. これは、高校課程の物理では実験との関連において、誤差、有効数字及び誤差の伝達などについてひと通りは触れられていることと比べるとその対照が一層明らかである¹⁾. 更に、第2として、高校課程の理科教育では、基本的原理や法則から演繹的に結論を導き出すことに主力が置かれ、化学が実験に立脚した学問であることが忘れられがちであることである. ところで、工学部の専門教育では実験値の取扱いについての理解が欠くことのできないものであることは言うまでもない.

そこで、化学研究室では、1年次生が履習する「化学実験」の授業において、学生が行った実験結果のデータ整理が正しく行われているか否かを判定するためのコンピューター・プログラムを作製して、学生が正しい答を出すまで反復学習させている.

誤差を含む数値をもとにして計算を行う場合に

* 一般教養助教授

昭和58年9月19日受理

は、まず数値が持つ誤差の大きさをどのように表わすかという問題と、計算結果に対するもとの数値の誤差の影響をどのように扱うかという問題が生ずる。これらを扱うにはいくつかの方法があるが、実験値の数が少ない場合には有効数字の桁数によって誤差の大きさを表わす方法が最もよく使われる²⁾。この方法では、それぞれ NA 桁と NB 桁の有効数字を持つ数値 A と B があるものとして A と B の間で乗除算を行った場合には、その結果が持つ有効数字の桁数は NA と NB のうちの小さい方の値とする。このような計算の手続きをコンピューター・プログラムとして記述するためには、計算の対象となる数値 A、B の他に、これらが持つ有効数字の桁数 NA と NB をも取扱う必要が生ずる。ところで、FORTRAN などの一般のプログラミング言語の中にはこのような処理を直接行うことのできる機能がないので、有効数字を考慮して数値計算を行うためのコンピューター・プログラムを作製した。

2. プログラムの概要

ここでは有効数字を考慮する数値計算として、浮動小数点演算による正の実数の乗除算を取り上げることとし、かつ、これらの実数は必ず小数部にも有効数字を持つものとする。1 で述べたような目的のためには、次に示すような機能を持ったプログラムが必要であり、これらをそれぞれ () 内に示した subroutine 名で呼ぶこととする。

- 1) 入力したデータの有効数字の桁数を求める (INPUT).
- 2) データの値を決められた有効数字の桁数で表示する (OUTPUT).
- 3) データの値を決められた桁数に丸める (RNDX).
- 4) ふたつのデータの値が決められた有効数字の桁数の範囲内で一致することを確認する (CHECK).

このうち、1) と 2) での処理の特徴は、データの内容を文字の並びとして扱うことであり、3) と

4) では、データの数値をひとつひとつの整数の並びとして扱っていることである。3) で行う数値を丸める方法のうちで最も良く使われているのは四捨五入である。ところが、これでは切上げの方に偏ってしまうので JIS では次のように規定している³⁾。

- 1) 丸める桁の数字が 4 以下のときは切捨て。
- 2) 丸める桁の数字が 6 以上のときは切上げ。
- 3) 丸める桁の数字が 5 のときは、丸める桁よりひと桁上位の桁の数字が、
 - a) 偶数のときには切捨て。
 - b) 奇数のときには切上げ。

これを考慮して subroutine RNDX では、これらのうちのどちらをも行うことができるように配慮されている。4) の subroutine CHECK では新たに入力されたデータ（有効数字 NA 桁の数値 A とする）とコンピューター内で確定されているデータ（有効数字 NB 桁の数値 B とする）との比較が行われる。ここで、処理する内容によってふたつの場合を別々に扱わなければならない。その第 1 は、プログラム利用者の入力した値が正しい計算結果と比較されるような場合であり、その第 2 は、円周率、分子量などのように計算を行う上で必要ではあるが、扱っている実験以外から得られる数値と利用者が入力した値とを比較する場合である。このうち、第 1 の場合では $NA=NB$ であることが必要であり、かつこの有効数字の桁数の範囲内で $A=B$ であることが必要となる。一方、第 2 の場合では、円周率、分子量などの値が十分に詳しくわかっているから、実験値の有効数字の桁数の最小値を NB とすると、 $NA \geq NB$ であれば十分であり、かつ NA 桁の範囲内で $A=B$ が成り立てばよい。これらを考慮して、CHECK の中ではこのどちらの場合をも処理することが可能となっている。

3. プログラム各論

以下に述べるプログラムの説明の中では、変数と配列に次のような意味を与えている。

IERR : 処理に異常が生じたことを表わす
 変数 (=0 正常, ≠0 異常).
ICNTL : 処理の流れを変えるための変数.
IDAT(I) : ある数値に対応する文字の並びを
 表わす配列.
IX, IY : ある数値 X または Y の有効数字か

ら成る整数.
NS : ある数値が持つ全体の有効数字の桁
 数.
NH : ある数値の整数部の有効数字の桁数.
NL : ある数値の小数部の有効数字の桁数.

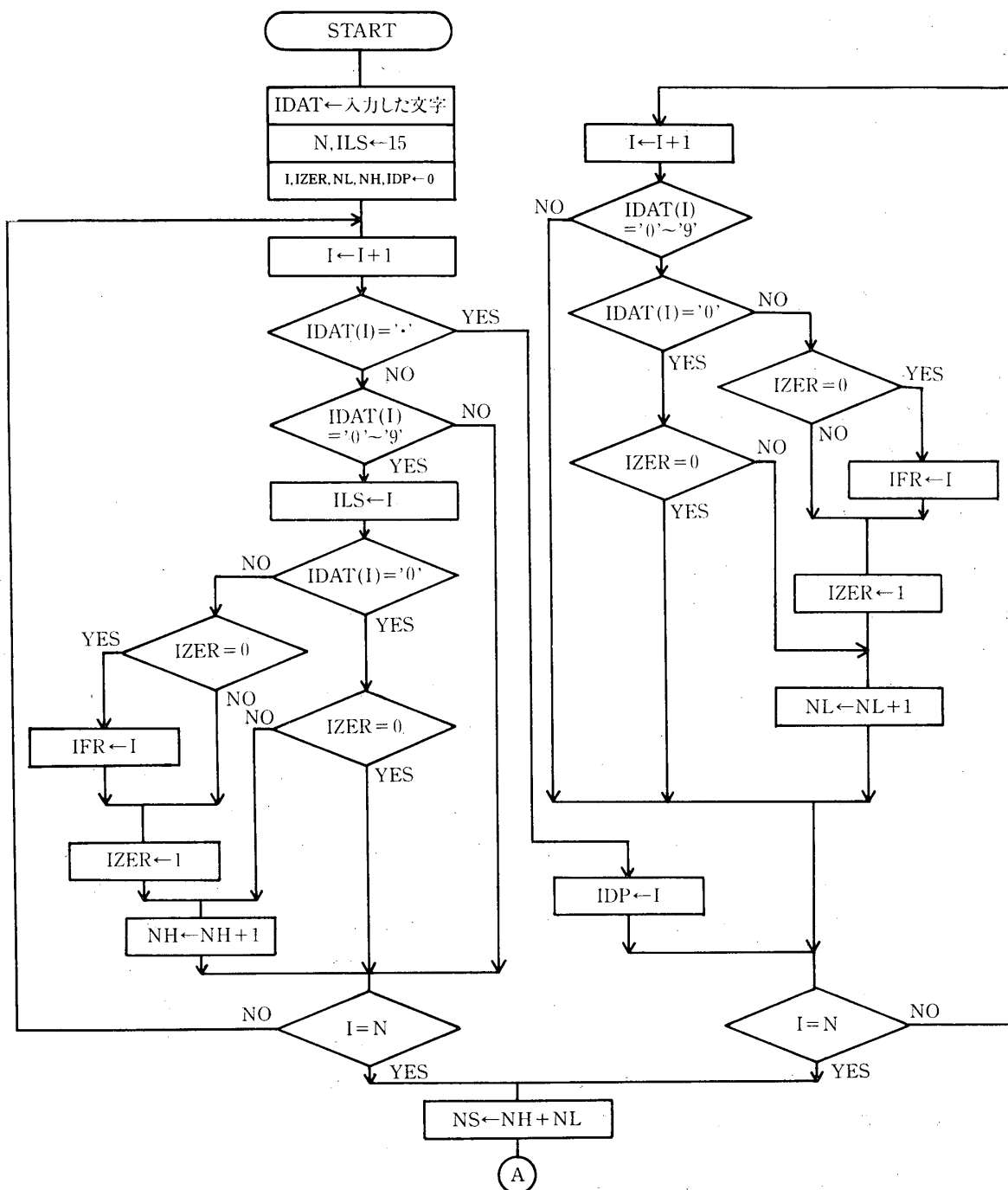


図 1a Subroutine INPUT の流れ図 (その 1, 有効数字の桁数の処理)

1) subroutine INPUT (X, NL, NH, NS, IERR)

図 1a 及び図 1b にプログラムの流れ図を示す。このプログラムではまずデータがひとつひとつの文字として配列 IDAT に入力され、これをもとにして NH, NL, 及び NS と、配列 IDAT の上で最上位の有効数字が表われる順番 IFS と整数部の 1 の位の数字が表われる順番 ILS 及び小数点が表われる順番 IDP をそれぞれ求める。次にこれらをもとにして配列 IDAT 上の文字の並びからこれに対応する実数 X 値変数の値を求める。ここで IZER は文字として 0 が入力されたときにこれを有効数字として扱うか否かを判定するための変数であり、IZER=1 のときには有効数字として扱い、IZER=0 のときには有効数字として扱わない。

2) subroutine OUTPUT (X, NS)

図 2 に流れ図を示す。ここでは、まず最上位の有効数字の位 MFX を求め、次に出力すべき実数型変数 X が持つ有効数字をひとつずつ配列 ICD に代入し、更に ICD の各要素に対応する文字を配列 IDAT 上に作り、最後に IDAT の内容を入力する。

3) subroutine RNDX (X, NS, ICNTL, IERR)

図 3 に流れ図を示す。このプログラムでは最初に丸めるべき値が入っている変数 X からその有効数字より成る整数 IX を求め、次にこの IX から丸めるべき桁の数字に対応する整数 II と、丸めるべき桁よりひと桁上位の数字の偶奇を表わす整数 IK (=0 のとき偶数, =1 のとき奇数) を求める。ICNTL は丸めを行う際に、JIS の規定によるか、四捨五入にするかを決定する変数である (ICNTL=0 のときは JIS の規定により、ICNTL=1 のときには四捨五入にする)。実際の丸めの操作は ICNTL, II 及び IK の値を判定して IX に対して行う。このとき、最上位の桁か

らも桁上りが生ずれば有効数字の桁数が増加することになるのでこれも考慮している。最後に IX の値から実数型データ X を得る。

この他に、実数型変数 X の整数部の桁数を与える subroutine function NFX (X) と、NS 桁の有効数字を持つ実数型変数 X の小数部の有効数字の桁数を与える subroutine function NLX (X,

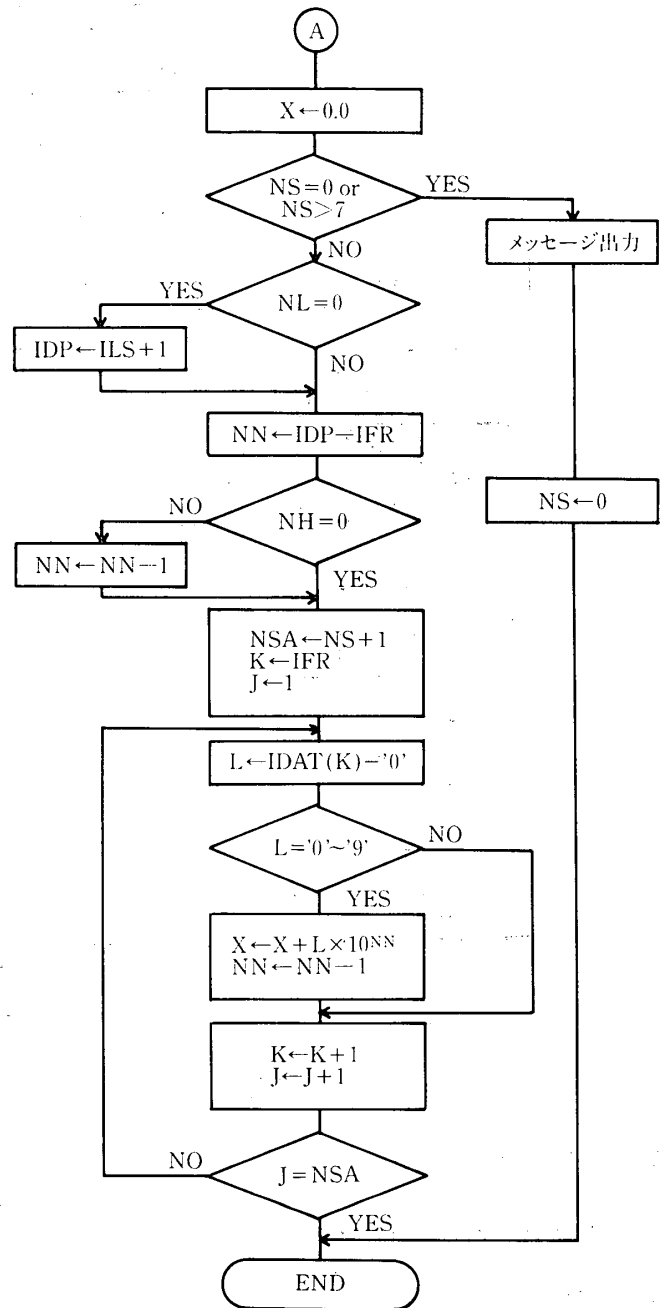


図 1b Subroutine INPUT の流れ図 (その 2, 文字の並び→数値への変換)

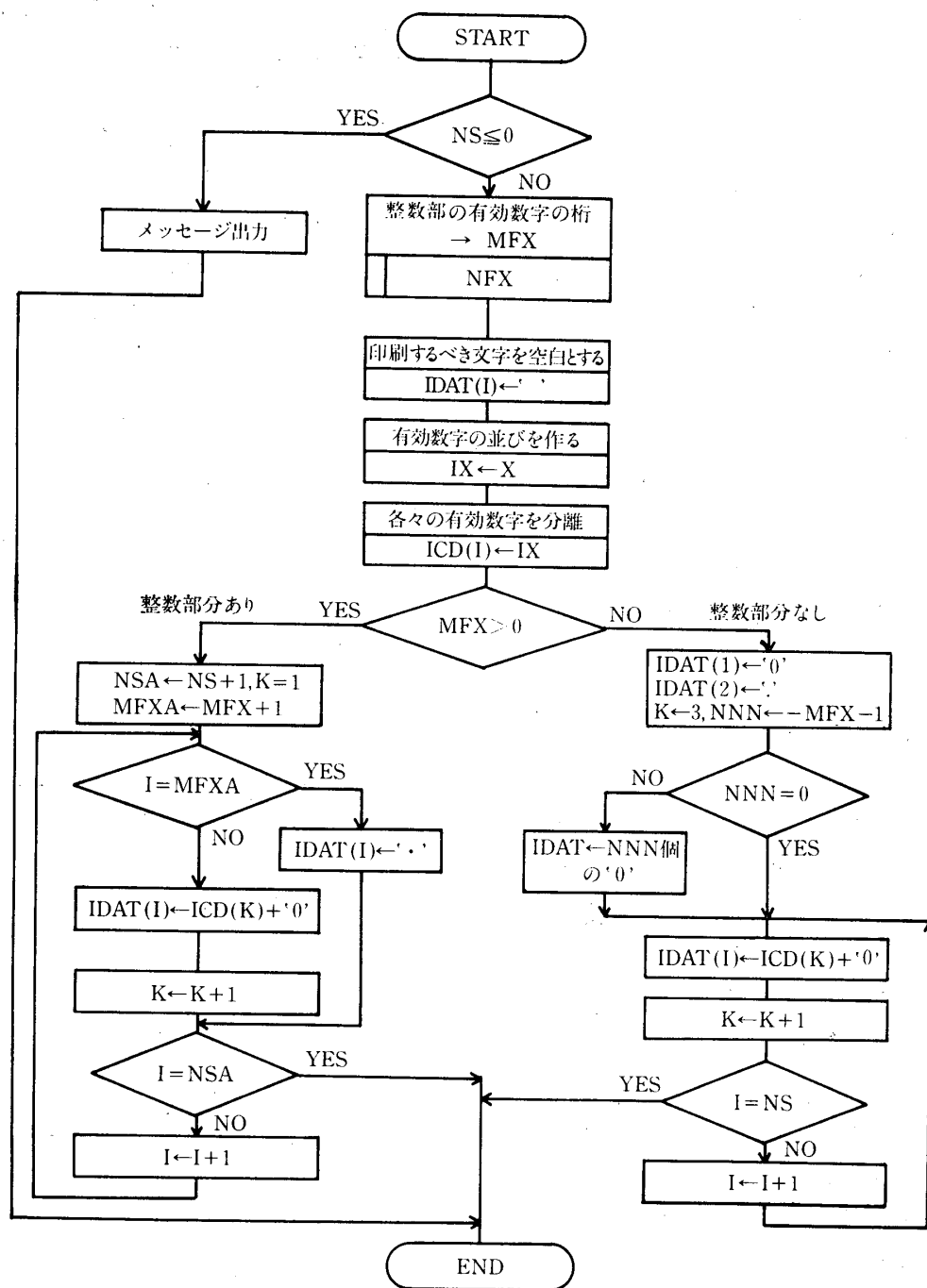


図 2 Subroutine OUTPUT の流れ図
(NNN: 小数部における有効数字ではない 0 の数)

NS) が用意されている。

4. プログラムの問題点と応用例

以上で述べたプログラムの中では整数型変数と実数型変数の間の相互の変換が必要となる。ところで、実際にある実数型変数を一度整数型変数へ

変換し、更に再び実数型変数へ変換すると、この値は始めの実数の値と一致するはずであるが、場合によっては両者の値が有効数字においても一致しなくなることがある。これは異なる型の変数へと変換されるごとに近似値がとられるからであり、特に実数型から整数型への変換では小数部分

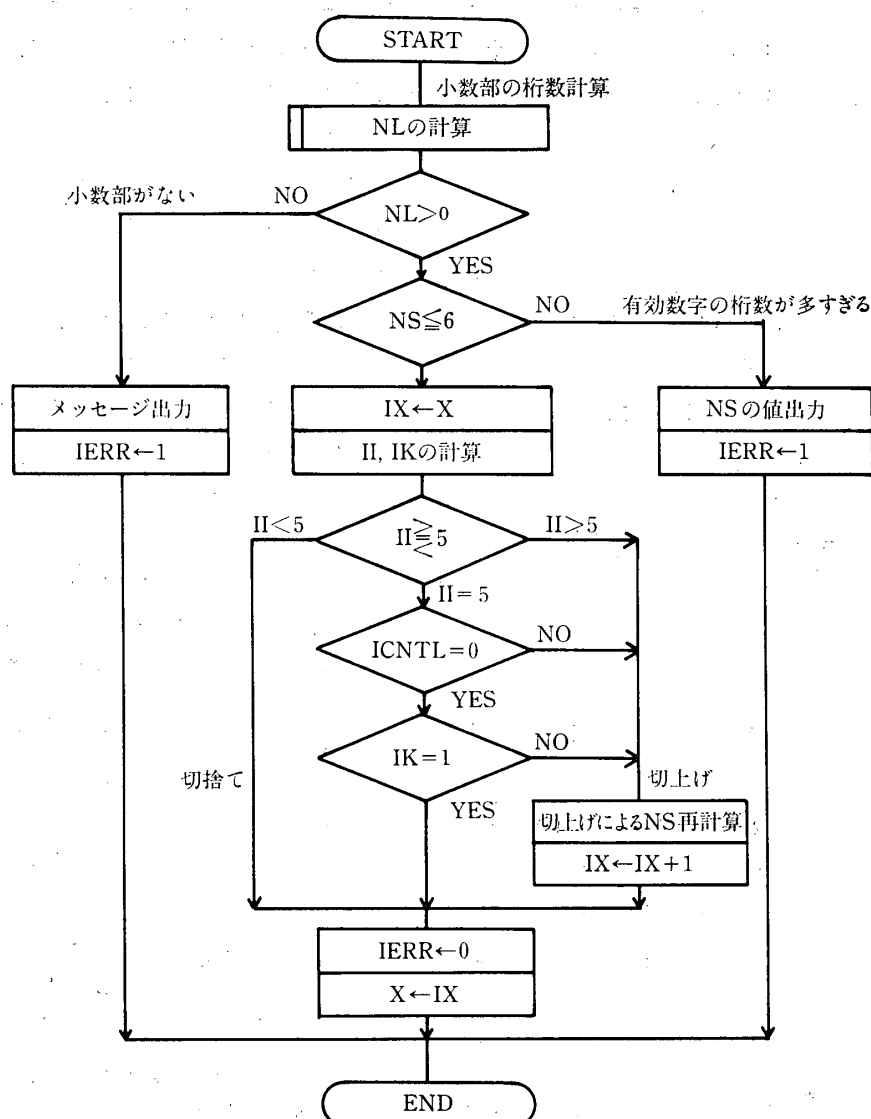


図 3 Subroutine RNDX の流れ図

の切捨てが行われるためであることがわかった。そのため、整数型から実数型への変換に際しては、過大な近似値となるようにした。ここで述べたプログラムは Microsoft 社の FORTRAN⁴⁾ によって書かれているが^{*}、この FORTRAN では単精度実数型データを 4 Byte として表わしており、仮数部は実質的に 24 bits の精度を持っている。従って、仮数部最下位 bit は最上位 bit に対して約 1.2×10^{-7} の比となる。そこで整数型から実数型に変換するときには、実数として得た値を

更に $(1 + 2 \times 10^{-7})$ 倍している。このような処理を加えているので、ここで述べたプログラムで数値計算を行う場合の計算精度は、FORTRAN による単精度実数による精度 (1×10^{-7}) より悪くなるが、6 桁の有効数字は保証することができる。これは、通常の実験によって得られるデータの整理を行う上では十分な精度である。

化学研究室では、この Microsoft 社の FORTRAN で書かれたプログラムを CP/M オペレーティングシステムの管理下で動くいくつかのマイクロコンピューターシステムで実行することができる。「化学実験」の授業では「中和滴定」と

* プログラムのソースリストは付録として最後に示してある。

「酸化還元滴定」のデータ整理の確認を行うためのプログラム、NEUTITR と RDXTITR を作り、実際に学生に利用させている。また、実際の授業ではコンピューターシステムの不足を補うためにカシオ FP-1100 システムも CP/M の下で動作させて使用している。

実際の FORTRAN ソースプログラム（付録参照）では、通常の FORTRAN の文法として許されるものだけを使っているので、これらのソースプログラムは他のシステムの FORTRAN でも実行可能であるが、1) 文字型データの内部表現として ASCII コードを仮定していること、2) 1 Byte 変数を使用可能と仮定していることには留意する必要がある。

5. ま と め

有効数字の取扱いを考慮した数値計算を実行するためのコンピューター・プログラムを FORTRAN 語により作製し、これらのプログラムが予期した機能を持つことが確かめられた。これらのプログラムは「化学実験」の授業で使用されている。このようにコンピューターを授業に取り入れることの効果などについては別の機会に発表した。

昭和 59 年度の高等学校卒業生から適用される新学習指導要領⁵⁾では、実験値の取扱いに關係す

る事項が殆んどなくなっており、大学教養課程における実験値の取扱いについての教育は更にその重要性を増すこととなるであろう。

最後に、プログラム作製の課程で貴重な指摘や助言をいただいた、鳥取大学教養部榊原正明助教授に深く感謝する。更に工学部 森紀夫修士からは「化学実験」で実験値のデータ整理を取りあげる際に貴重な示唆をいただいた。また、化学研究室の大学院、卒研生諸君には、これらのプログラムを授業で使う場合に、新入生にコンピューターの操作を教えていただいた。ここに重ねて感謝の意を表わす次第である。

参 考 文 献

- 1) 「高等学校学習指導要領」文部省（昭和 45 年 10 月）。ただし、これは昭和 58 年度卒業生までに適用されているものであり、昭和 59 年度卒業生からは新しい指導による授業内容となる。
- 2) R. A. Day and A. L. Underwood, 鳥居泰男, 康智三訳「定量分析化学」（改訂版）培風館（昭和 57 年）。
- 3) JIS Z 8401-1961「数値の丸め方」。ただし、この規定の文章は大変わかりにくい。同じ内容が次の文献に要領よくまとめられている。浅田誠一, 内出 茂, 小林基宏「図解とフローチャートによる定量分析」技報堂出版（1981）。
- 4) Microsoft FORTRAN-80 Reference Manual, MicroSoft（1981）。
- 5) 「高等学校学習指導要領解説（理科編, 理数編）」文部省（昭和 54 年 4 月）。これは昭和 57 年度の高校入学から適用されている。

付 録 （プログラムのソースリスト）

```

C-----
C      Subroutine package for the treatment of significant figures
C      Copyright by M. Uyemura ( Tokyo Institute of Polytechnics )
C      Sept. 15, 1983
C-----
C      subroutine INPUT
C
C      Last revision on Nov. 26, 1982
C
C      DATA X O NYURYOKU SURU
C
C      SUBROUTINE INPUT(X,NL,NH,NS,IERR)
C      INTEGER*1 IDAT(15),DPINT,D9,D0,IA
C      DATA DPINT,D9,D0,IA/' ','9','0','A' /

```

C

```

1 IERR=0
2 READ(1,100) IDAT
  IF(IDAT(1).EQ.IA) GO TO 92
  N=15
  I=0
  IZER=0
  NL=0
  NH=0
  IDP=0
  ILS=15
10 I=I+1
  IF(IDAT(I).EQ.DPINT) GO TO 21
  IF(IDAT(I).LT.D0.OR.IDAT(I).GT.D9) GO TO 18
  ILS=I
  IF(IDAT(I).NE.D0) GO TO 12
  IF(IZER.NE.0) GO TO 14
  GO TO 18
12 IF(IZER.EQ.0) IFR=I
  IZER=1
14 NH=NH+1
18 IF(I.LT.N) GO TO 10
  GO TO 30
21 IDP=I
  GO TO 23
22 I=I+1
  IF(IDAT(I).LT.D0.OR.IDAT(I).GT.D9) GO TO 23
  IF(IDAT(I).NE.D0) GO TO 24
  IF(IZER.NE.0) GO TO 26
  GO TO 30
23 IF(I.GE.N) GO TO 22
24 IF(IZER.EQ.0) IFR=I
  IZER=1
26 NL=NL+1
  GO TO 23
30 NS=NL+NH
  X=0.
  IF(NS.GT.6) GO TO 90
  IF(NS.EQ.0) GO TO 50
  IF(NL.EQ.0) IDP=ILS+1
  NN=IDP-IFR
  IF(NH.NE.0) NN=NN-1
  NSA=NS+1
  K=IFR
  DO 40 J=1,NSA
    L=IDAT(K)-D0
    IF(L.LT.0.OR.L.GT.9) GO TO 42
    X=X+FLOAT(L)*10.**NN
    NN=NN-1
42 K=K+1
50 CONTINUE
  X=0.100000002E+01*X
  RETURN

```

C

```

90 WRITE(1,102) NS
  NS=0
  RETURN
92 WRITE(1,104)
  IERR=1
  RETURN

```

C

```

100 FORMAT(15A1)
102 FORMAT(' ????? Yuukousuji wa ',I4,' -keta to naru'
  * ' ***** '/')
104 FORMAT('// ***** Aborted *****')
  END

```



```

C-----
C      Last revision on Nov. 26, 1982
C
C      DATA X O YUUKOOSUJI NS KETA DE SHITSURYOKU SURU
C
C      SUBROUTINE OUTPUT(X,NS)
C      INTEGER*4 IX,IC
C      INTEGER*1 IDAT(15),ICD(10),SPACE,DP,D0
C      DATA SPACE,DP,D0/' ','.', '0'/
C
C      N=15
C      IF(NS.LE.0) GO TO 90
C      MFX=NFX(X)
C
C      DO 10 I=1,N
C      IDAT(I)=SPACE
C 10 CONTINUE
C
C      NL=NLX(X,NS)
C      IX=X*10.**NL
C      IC=IX
C
C      DO 20 I=1,NS
C      IN=NS-I+1
C      ICD(IN)=IC-(IC/10)*10
C      IC=IC/10
C 20 CONTINUE
C
C      IF(MFX.GT.0) GO TO 35
C      IDAT(1)=D0
C      IDAT(2)=DP
C      K=3
C      NNN=-MFX-1
C      IF(NNN.EQ.0) GO TO 32
C      DO 30 I=1,NNN
C      IDAT(K)=D0
C      K=K+1
C 30 CONTINUE
C 32 CONTINUE
C      DO 34 I=1,NS
C      IDAT(K)=ICD(I)+D0
C      K=K+1
C 34 CONTINUE
C      GO TO 50
C
C 35 NSA=NS+1
C      MFXA=MFX+1
C      K=1
C      DO 40 I=1,NSA
C      IF(I.EQ.MFXA) GO TO 45
C      IDAT(I)=ICD(K)+48
C      K=K+1
C      GO TO 42
C 45 IDAT(I)=DP
C 42 CONTINUE
C 40 CONTINUE
C
C 50 CONTINUE
C
C      WRITE(1,102) IDAT
C      CALL TWAIT(2)
C      RETURN
C
C 90 WRITE(1,100) NS
C      RETURN
C
C 100 FORMAT(' ????? Error in OUTPUT *** NS=',I4)
C 102 FORMAT(' ***** Kotae wa ***** ',15A1)
C      END

```

```

C-----
C
C      Last revision on Sept. 15, 1983 by M.U.
C
C      SUBROUTINE RNDX(X,NS,ICNTL,IERR)
C      INTEGER*4 IX,JX,KX,LX
C
C          NL=NLX(X,NS)
C          IF(NL.LE.0)                GO TO 90
C          IF(NS.GT.6)                GO TO 92
C          XF=10.**(NL)
C          XXF=X*XF
C          IX=XXF
C          JX=XXF*10.
C          KX=XXF/10.
C          KK=IX-KX*10
C          II=JX-IX*10
C          IK=KK-(KK/2)*2
C          IF(II.GT.5)                GO TO 10
C          IF(II.LT.5)                GO TO 20
C          IF(ICNTL.EQ.0.AND.IK.NE.1) GO TO 20
10      CONTINUE
C
C      Readjust NS
C
C          XX=IX
C          NSX=NFX(XX)
C          LX=10**NSX-1
C          IF(IX.EQ.LX)                NS=NS+1
C
C          IX=IX+1
C
C      20 CONTINUE
C
C      40 IERR=0
C          XX=IX
C          X=XX/XF
C          X=0.10000002E+01*X
C          RETURN
C
C      90 WRITE(1,102)
C          IERR=1
C          RETURN
C
C      92 WRITE(1,104) NS
C          IERR=1
C          RETURN
C
C      102 FORMAT(' !!!!! Kotae wa seisuu to narutame',
C          *      '      'yuukousuuji wa aimai ni naru *****')
C      104 FORMAT(' ?????? Yuukousuji wa 'I3,'-keta to naru *****')
C      END

```

```

C-----
C
C      Last revision on Sept. 15, 1983      by M.U.
C
C      Check input data
C
C      ICNTL : Controls processing
C
C      SUBROUTINE CHECK(A,NSA,B,ICNTL,IERR)
C      INTEGER*4 IA,IB
C
C      IERR=0
C
C      Data input from keyboard
C
C      CALL INPUT(B,NLB,NN,NSB,IERR)
C
C      IF(IERR.NE.0)          GO TO 96
C      IF(NSB.EQ.0)           GO TO 92
C      IF(ICNTL.EQ.0.AND.NSB.NE.NSA) GO TO 94
C      IF(ICNTL.NE.0.AND.NSB.LT.NSA) GO TO 94
C      IF(ICNTL.EQ.0)         NSC=NSA
C      IF(ICNTL.NE.0)         NSC=NSB
C
C      Round input data
C
C      CALL RNDX(A,NSC,1,IERR)
C
C      Compare input data with reference
C
C      NL=NLX(A,NSC)
C      AF=10.** (NL)
C      IA=A*AF
C      IB=B*AF
C      IF(IA.NE.IB)           IERR=1
C      IF(IERR.EQ.0)          RETURN
C
C      Error
C
C
C      90 WRITE(1,100)
C      RETURN
C      92 NSAA=NSA+ICNTL
C      CALL RNDX(A,NSAA,1,IERR)
C      CALL OUTPUT(A,NSAA)
C      IERR=2
C      RETURN
C      94 WRITE(1,102)
C      IERR=3
C      RETURN
C      96 IERR=10
C      RETURN
C
C      Format
C
C      100 FORMAT(' ???? Input data no machigae'//)
C      102 FORMAT(' ???? Yuukousuji no machigae'//)
C      END
C

```